



Linux

CPT-201

<http://cpt201.hubner.org/>

Tests

- Week 6 - Oct 03 - Test 1 - GNU/Linux, Licensing, RedHat, CentOS, Canonical, Ubuntu, Linus Torvaldes, Richard Stallman, Mark Shuttleworth
- Week 9 - Oct 29 - Test 2 - Linux Commandline
- Week 12 - Nov 14 - Test 3 - System Admin
- Week 15/16 - Dec 5/12 - Final

Practice At Home with a LiveCD

The screenshot displays a LiveCD environment. The top bar shows the system menu (Applications, Places, System) and the date/time (Wed Oct 10, 5:32 PM). The web browser window is titled "CPT-201 Fall 2012 Main Page - Classes - Iceweasel" and shows the URL `cpt201.hubner.org/wiki/CPT-201_Main_Page`. The page content includes a sidebar with a penguin mascot, a navigation menu, and a search box. The main content area displays the "CPT-201 Fall 2012 Main Page" with tabs for article, discussion, edit, and history. A terminal window titled "amnesia@amnesia: ~" is open in the foreground, showing the output of the `ls(1)` command. The terminal output includes the command `ls - list directory contents`, the synopsis `ls [OPTION]... [FILE]...`, and the description: "List information about the FILES (the current directory by default). Sort entries alphabetically if none of -cftuvSUX nor --sort." The terminal also shows the mandatory arguments to long options and the manual page reference "Manual page ls(1) line 1".

Applications Places System Wed Oct 10, 5:32 PM

CPT-201 Fall 2012 Main Page - Classes - Iceweasel

File Edit View History Bookmarks Tools Help

CPT-201 Fall 2012 Main Page - C...

cpt201.hubner.org/wiki/CPT-201_Main_Page

Log in / create account

CPT-201

article discussion edit history

CPT-201 Fall 2012 Main Page

(Redirected from CPT-201 Main Page)

CPT-201 Spr 2012 (Li

- Main Page
- Syllabus
- Homework
- Tests
- Help

Search

Go Search

Toolbox

- What links here
- Related changes
- Upload file
- Special pages
- Printable version
- Permanent link

Linux [edit]

- Main Page
- Syllabus
- Homework
- Tests
- Help

amnesia@amnesia: ~

File Edit View Terminal Help

LS(1) User Commands LS(1)

NAME

ls - list directory contents

SYNOPSIS

ls [OPTION]... [FILE]...

DESCRIPTION

List information about the FILES (the current directory by default). Sort entries alphabetically if none of -cftuvSUX nor --sort.

Mandatory arguments to long options are

Manual page ls(1) line 1

Practice At Home with a VM

<http://www.VirtualBox.org/>



*VirtualBox is free and
available for Windows, Linux, MacOS X*

Fix Up Sudo

- We're going to start relying on Sudo, so lets make sure it works for everybody...
- You should be able to do this command:

```
sudo su -
```

- If you can't, make sure:
 - Your **Username** is part of the **wheel** group
 - **/etc/sudoers** contains:

```
%wheel  ALL=(ALL)  NOPASSWD:  ALL
```

Command Chaining

- Commands can be chained together using the power of the **pipe** (**|**).
- The output of one command can be fed, via **pipe** (**|**), to another command.

```
echo "Hello" | wc
```

```
ps auwx | grep sh
```

```
ls | sort | less
```

Aliases

- Aliases are shortcuts, or overrides, that refer to other commands.
- So the command “look” could be set to be the same as “ls -al”.

```
alias 'look=ls -al'
```

- For our Windows friends moving to Linux, the command “dir” could be set to be the same as “ls”

```
alias 'dir=ls'
```

Aliases

- Aliases can be used to add options to commands. So an alias can be an alias of a real command

```
alias 'ls=ls --color=auto'
```

```
ls -al
```

```
alias 'ls=ls --color=auto'
```

```
ps auwx | grep sh
```


Aliases

- Aliases can be used to override existing commands

```
alias 'ls=yes "Uh uh  
uh, you didnt say the  
magic word"'
```

```
ls
```

BASH \$Variables

- BASH has many variables that govern how it acts.
- We learned about **PS1** last week..

```
export PS1="\u@\h:\W# "
```

BASH \$Variables

- We've talked about the bash history that gets displayed when you type the command:

history

- The bash history is normally stored in

~/ .bash_history

- The bash history log file can be overwritten with the **HISTFILE** variable

HISTFILE=~/ .history.log

BASH \$Variables

- If you don't want your commands recorded in the history file, set the **HISTFILE** variable to nothing

HISTFILE=

```
echo "hello"
```

```
ls /
```

```
history
```

```
tail ~/.bash_history
```

BASH \$Variables

- The EDITOR variable determines which text editor various commands that need a text editor use.

```
EDITOR=nano
```

```
visudo
```

```
EDITOR=gedit
```

```
visudo
```

BASH \$Variables

- Variables are easily assigned using the method:

`VariableName="Value"`

- Example:

```
FAUST="Likes to Party"
```

```
echo "Faust $FAUST"
```

BASH \$Variables

- You can capture the output of commands and then re-use it.

```
THEDATE=$(date)
```

```
echo $THEDATE
```

BASH scripts

- Last week we colorized the penguin.
- This penguin is actually a bash script.
- A Bash script is a text file containing a bunch of commands.



Bash Scripts

- Bash is in essence a command interpreter, making it a programming language.
- To call a bash script:

```
bash penguin.sh
```

- Or

```
chmod +x penguin.sh  
./penguin.sh
```

HashBang / SheBang

- If a script is executable (+x) then the top line will be used as the interpreter for the script.
- The interpreter line is called a **HashBang** or **SheBang**.

! /bin/bash

HashBang / SheBang

- Bash: `#!/bin/bash`
- Ruby: `#!/usr/bin/ruby`
- Python: `#!/usr/bin/python`
- Perl: `#!/usr/bin/perl`
- PHP: `#!/usr/bin/php`

- *Pull it from the environment:*

`#!/bin/env ruby`

Create a Ruby Script

```
#!/usr/bin/ruby  
puts "Hello"
```

Install Ruby if you don't have it:

```
sudo yum -y install ruby
```

```
chmod +x myRubyScript.rb  
./myRubyScript.rb
```

Create a Bash Script

```
#!/bin/bash
```

```
THEDATE=$(date)
```

```
echo "The Date is: $THEDATE"
```

```
chmod +x myBashScript.sh  
./myBashScript.sh
```

Add Colors to the Script

```
(21:23:13) [devon@t510-dhubner:~] ./hello.sh  
The Date is: Wed Oct 10 21:23:14 EDT 2012
```

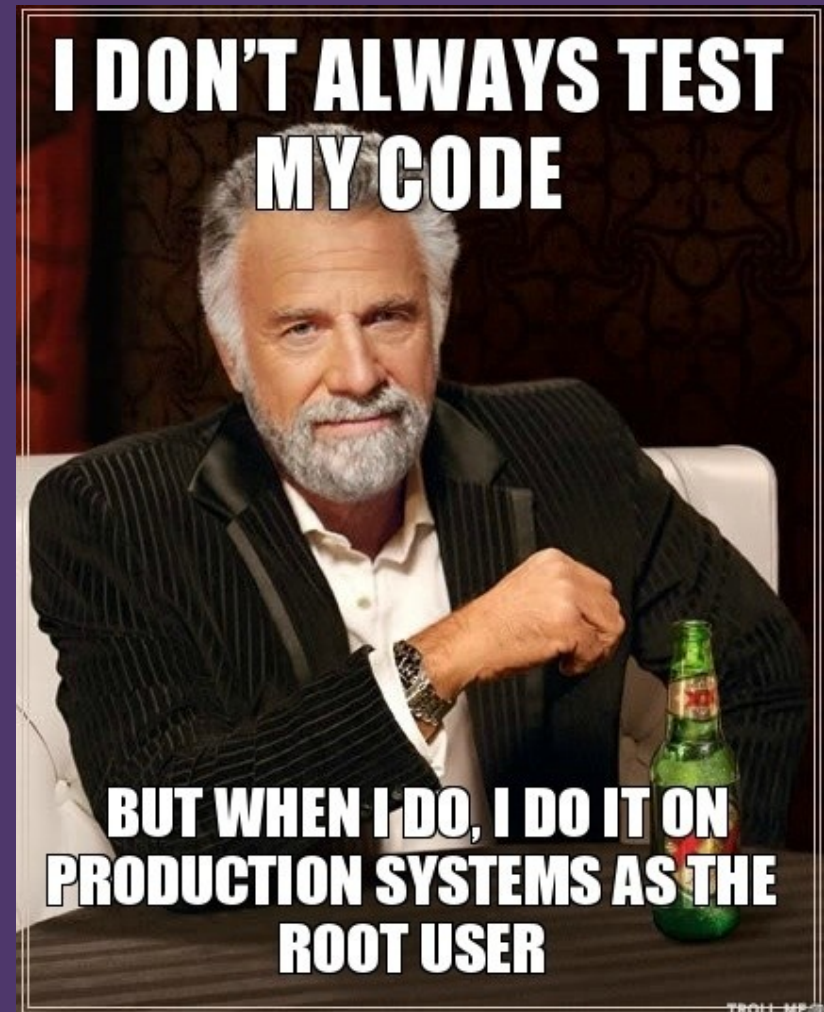
```
export NORMAL="\e[0m"  
export RED="\e[31;1m"  
export GREEN="\e[32;1m"  
export YELLOW="\e[33;1m"  
export BLUE="\e[34;1m"  
export CYAN="\e[36;1m"
```

Test 2 Review

- Week 6 - Oct 03 - Test 1 - GNU/Linux, Licensing, RedHat, CentOS, Canonical, Ubuntu, Linus Torvaldes, Richard Stallman, Mark Shuttleworth
- Week 9 - Oct 29 - Test 2 - Linux Commandline
- Week 12 - Nov 14 - Test 3 - System Admin
- Week 15/16 - Dec 5/12 - Final

root

- User: root
- Filesystem: /
- Home Directory: /root



Use wget to download files

wget

**http://192.168.
7.7/penguin.txt**

Use **yum** and **apt** to install apps

```
yum -y install git
```

```
apt-get install -y git
```

Finding Help!

- `man <command name>`
- `man dd`
- `man cp`
- `man ls`
- `man head`
- `man tail`
- `man cat`

Make Directories

```
mkdir -p a/b/c/d/e/f/g
```

```
ls -R a
```

Tilde (~)

```
ls ~
```

```
ls /home/username
```

```
ls ~username
```

Tail, Head, Cat

tail filename

head filename

cat filename

Homework

- http://cpt201.hubner.org/wiki/CPT-201_Fall_2012_Homework#Class_7_-_2012.10.10
- Please answer **all** the questions.
- Answers are due by the start of **class 9** on **October 24th**.
- You can try out these commands at home using a **LiveCD** like the **System Rescue CD**. You may also burn one of the other LiveCD's available, like **Tails**.

Permissions

- +x Add Executable
- +r Add Read
- +w Add Write

- -x Remove Executable
- -r Remove Read
- -w Remove Write

User - Group - Other

- u=rwx
- g=rx
- o=rx
- `chmod u=rwx,g=rx,o=rx hello.sh`